Research Article

Energy Theft Detection in an Edge Data Center Using Deep Learning

Guixue Cheng ¹, Zhemin Zhang ¹, Qilin Li ², Yun Li,³ and Wenxing Jin¹

¹College of Computer and Science and Technology, Shanghai University of Electric Power, Shanghai 200090, China

²Metering Center of Sichuan Electric Power Corporation, Chengdu 610045, China

³School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Correspondence should be addressed to Qilin Li; li_qi_lin@163.com

Received 16 March 2021; Revised 26 April 2021; Accepted 28 June 2021; Published 8 July 2021

Academic Editor: Junming Huang

Copyright © 2021 Guixue Cheng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the development of smart grid information physical systems, some of the data processing functions gradually approach the edge layer of end-users. To better realize the energy theft detection function at the edge, we proposed an energy theft detection method based on the power consumption information acquisition system of power enterprises. The method involves the following steps. In the centralized data center, K -means is used to decompose a large amount of data into small data and then input and train neural network parameters to realize feature extraction. We design a neural network named DWMCNN, which can extract features from the day, week, and month and can extract more accurate features. In the edge data center, the random forest (RF) algorithm is used to classify the extracted features. The experimental results show that the clustering method accords with the idea of edge computing-distributed processing and improves the operation speed and that the feature extractor has good convergence performance. In addition, compared with the methods based on various classifiers, this method has higher accuracy and lower computational complexity, which is suitable for the deployment of edge data centers.

1. Introduction

In recent years, the industrial Internet of Things, especially smart grids, has developed rapidly. In the smart grid, the advanced metering infrastructure (AMI) [1], combined with the Internet of Things technology and artificial intelligence technology, can obtain historical and real-time power consumption data from meters deployed in users' homes to realize emergency analysis and transient stability simulation [2, 3].

However, smart meters are vulnerable to network physical attacks in smart grids due to their insecure distributed distribution and physical environment, resulting in energy theft. The loss caused by energy theft belongs to the nontechnical loss of electric power loss. By attacking smart meters and installing electricity stealing modules for meters, this behavior can wiretap, damage, and tamper with meter readings, resulting in a significant income loss of energy enterprises and even endangering public safety (such as fire

or electric shock). The electricity stealing rate in developing countries is quite high, reaching 30%. In India, energy theft costs as much as \$4.5 billion a year [4].

Traditional energy theft detection mainly relies on the electric power enterprise to send technical personnel to read the electricity meter on a regular basis and then record, count, and analyze the data for manual discrimination. There are also methods of using camera monitoring to prevent energy theft. However, this method consumes the human and material resources of power enterprises and cannot detect the energy theft realized by advanced attack means. At present, the most commonly used method is combined with smart grid detection. Smart meters upload the collected data to a centralized data processing center, and then the centralized data processing center detects the theft through an intelligent algorithm. However, the widely deployed smart meters and a large amount of power consumption data pose challenges to the centralized data center processing mode. To save on the energy consumption of

nodes and reduce unnecessary data transmission, deploying edge data processing center with a data processing function at the user edge has become a new detection mode of energy theft. In this mode, the user's electricity data do not need to be uploaded to the centralized data center, which reduces the upload bandwidth.

Therefore, this paper aims to design a novel detection method to solve the above problems. We proposed a neural network model that is suitable for the deployment of edge devices and conforms to the daily, weekly, and monthly power consumption features of users to learn the power consumption data and identify the energy thieves.

The rest of the paper is organized as follows. Section 2 summarizes the related literature. Section 3 introduces the various parts of the model and the general process. We describe the characteristics of power consumption data in Section 4. We introduce the DWMCNN feature extraction model in Section 5. Then, we give the experimental results in Section 6. Finally, we summarize the thesis in Section 7.

2. Related Works

Through much research, we realized several detection technologies for energy theft. Researchers divide energy theft detection systems into three basic methods: state-based detection, game theory-based detection, and classification-based detection. Using upgraded devices and sensors in condition-based detection can improve the accuracy of energy theft detection. In [5], the authors designed a system that can conveniently detect and shield electricity stealing. The whole smart meter sensor is equipped with programmable logic controller (PLC) control and supervisory control and data acquisition (SCADA) monitoring. Energy theft detection occurs through a sensor that works during any illegal use of electricity. The main limitations of the detection system are vulnerability, high cost of hardware equipment, and high maintenance cost. The detection method based on game theory is suitable for analyzing a large amount of data. Reference [6] proposed a detection method based on game theory to find an optimal solution that is based on the formulation of various potential strategies. In this process, the greatest challenge is to calculate the utility function among distributors, regulators, and thieves. The method based on classification mainly uses a machine learning algorithm to establish a classification model and analyze the daily power consumption mode of users. The classification model includes decision trees (DTs), random forests (RFs), support vector machines (SVMs), and neural networks (NNs), and so on.

A classifier based on machine learning is used because the power consumption data are usually in the form of one dimension and time series. There have been many new studies [7–17], and the support vector machine (SVM) classifier is the most common method. In addition, there are studies [18–20] that used artificial neural networks to detect energy theft. The accuracy of these studies in the detection of energy theft is very low. Additionally, the features extracted by traditional machine learning feature extraction methods cannot successfully achieve effective energy theft detection.

In [21], the author trained an SVM algorithm model and rule engine algorithm using energy consumption data from customers with different time interval values. The different models proposed in the study achieved high success rates of 85.5% and 92%. The work in [22] proposed a convolutional neural network-long short-term memory (CNN-LSTM) model to detect energy theft in the smart grid. Due to the imbalance of data distribution, data generation technology was used in energy theft detection. This increases the amount of theft data to the same level as normal users. Although the experimental accuracy was 89%, in practice, energy thieves are often far less accurate than normal users. The work in [23] developed a new method to detect and identify energy theft in distribution systems using a multilayer perceptron artificial neural network (MP-ANN) algorithm. They successfully classified malicious users and normal users with an average accuracy of 93.4%. The work in [24] tried to identify customers who steal electricity by using smart meters through two different algorithms based on the linear regression method. Moreover, the study in [25] proposed a combination of convolutional neural network (CNN) and long short-term memory (LSTM) structures in which a model is used for short-term load forecasting and detection. Compared with other methods, the proposed model performed quite well.

There are also some methods that use clustering to detect anomalies. In [26], the density-based application spatial clustering and noise (DBSCAN) algorithm was used to detect and diagnose abnormal building operation patterns. In [27], a fuzzy clustering detection algorithm based on c-means was proposed. The Euclidean distance between customer consumption and regular profile was calculated and used to measure anomaly degree.

However, whether using the machine learning method or deep learning method, most of the related research cannot be well combined with edge computing platforms, cannot achieve high efficiency and low computational complexity, and is unsuitable for edge platform deployment. Therefore, this paper is committed to proposing a power theft detection model suitable for deployment in edge nodes, which can save on bandwidth and detect energy theft.

3. Proposed Method

In this section, we discuss the basic structure of the energy theft detection model based on edge computing and the basic process of energy theft detection. The energy theft detection model based on edge computing is composed of users, field terminals, edge data centers, and centralized data centers. Its specific structure is shown in Figure 1. This is described in detail in the next section.

3.1. System Model

Users: each user has a smart meter that can connect smart devices at home to aggregate their energy consumption. Users can be roughly divided into residential users, low-voltage general industrial and commercial users, small and medium-sized special transformer

FIGURE 1: Power-stealing monitoring system for edge computing.

users, and large-scale special transformer users. Each type of user can be divided into several categories according to their electricity consumption habits.

Field terminal (FT): this includes centralized meter reading terminals and special transformer acquisition terminals that can collect the data collected by the electric energy meter installed in the user's home, conduct a small amount of simple data processing, and monitor the operation status of the electric energy meter.

Edge data center (EDC): an edge data center is composed of sensors and processors with certain computing power. It can be deployed near the components of the distribution network and can carry out simple calculation tasks.

Centralized data center (CDC): a centralized data center has powerful computing power and a large number of computer resources. One panel is configured in medium and low-voltage distribution stations, distribution stations, and other places. After receiving the data collected by the field terminal, the edge gateway can carry out the simple calculations and then send them to the centralized data processing center.

3.2. System Flow. The computing power of the edge data center is not as strong as that of the centralized data center, so it cannot complete the task of energy theft detection only at the edge. However, it will take many resources to upload to the data center. When we design our energy theft detection scheme, we will fully consider the advantages and disadvantages of the edge end and central end of the equipment and design a set of power-stealing detection schemes that can make the edge data center and the centralized data center cooperate with each other and give full play to their respective advantages. The specific process of the scheme is shown in Figure 2.

- (1) The user's historical electricity consumption data (at least one year's data) collected by smart meters are uploaded to the CDC through the EDC. In the CDC,

the K -means algorithm is used to cluster the data sets for different electricity consumption habits. The K -nearest neighbor (KNN) model can also be trained to classify the data after clustering.

- (2) The clustering data are sent to different EDCs, and the model parameters of the CNN feature extractor are trained in the EDC. After training, the network parameters have good universality and persistence and do not need to be updated again in a short time. It should be noted that the given user history data $x \in R^{M \times N}$, where M and N denote the number of samples and the length of observation, respectively. It needs to be divided into a training set and test set $\{x_{\text{train}}, x_{\text{test}}\}$:

$$M_{\text{train}} \approx (60\% \sim 80\%) \times M, \quad (1)$$

$$M_{\text{test}} = M - M_{\text{train}}. \quad (2)$$

The CNN feature extractor is trained on x , and the weight learned is used as the feature of the input data of the classifier, which is recorded as $C(x)$. $C(x)$ is also segmented according to formula (1).

- (3) EDC uses $C(x)$ as the input data of the RF classifier to train the RF classifier model. The obtained model is stored in the EDC locally. When the parameters of the model need to be changed, the model is directly modified in EDC to reduce the bandwidth pressure of data upload.
- (4) To evaluate the model, the test set is used to check the performance of the whole scheme. When new user data are uploaded from the EDC, the power consumption type of the user is determined according to a KNN algorithm in the nearest EDC and transmitted to the corresponding EDC. According to the trained CNN-RF model, whether the user is an energy thief is determined.

This process includes three steps: (1) a large amount of data cleaning and preprocessing, (2) CNN feature extraction model training, and (3) RF classification model training. The

FIGURE 2: Power-stealing monitoring flow chart for edge computing.

first part of the content, a large amount of data with high computational complexity, needs to be processed at a centralized data center. In steps 2 and 3, the amount of data is small, and anomaly detection can be realized at the edge data center in less time.

4. Analysis of Electricity Consumption Data of Users

Many countries or distribution companies record real electricity consumption daily and regularly to investigate consumers' electricity consumption behavior. To determine the difference between normal users and energy thieves, we select a dataset released by the State Grid Corporation of China (SGCC) for analysis. This dataset contains the power consumption data of 42,372 power users over 1035 days.

We randomly selected the electricity consumption data of a typical normal user and a typical power-stealing user to draw, to determine the difference between them. Figure 3 shows a comparison of the daily electricity consumption data of two kinds of users (partial dates). It is obvious that the daily electricity consumption of the power-stealing users fluctuates significantly, while the daily electricity consumption of normal users fluctuates slightly. If we plot in two dimensions by week and month, we can obtain the difference between normal users and power-stealing users.

Using the method of two-dimensional drawing by week and a Pearson correlation coefficient, Zheng [28] found that the power consumption data of normal users have a strong correlation, but this only proved the regularity of power consumption of users with the week as a unit. We found that not only the user's electricity consumption data but also monthly correlation characteristics have weekly correlation characteristics.

Figure 4 shows the monthly electricity consumption curve of normal users, and Figure 5 shows the monthly electricity consumption curve of energy thieves. We can find the difference. The annual electricity consumption of power-stealing users is irregular, while the electricity consumption curve of normal users is periodic. The average power consumption of the three years reaches a peak in July every year,

and the power consumption of other months is low. (For the sake of fairness, the normal users and power-stealing users we choose are small and medium-sized households with an annual power consumption of less than 20,000 kWh.). If the method of analyzing one-dimensional time series data is used to analyze the user's electricity consumption data, then it is often difficult to obtain the characteristics of the regularity of the user's electricity consumption data. Many traditional data analysis methods, such as SVM and simple artificial neural networks (ANNs), cannot be directly applied to power consumption data due to their computational complexity and limited generalization ability.

Some scholars [28] considered the periodicity of power consumption data in the detection, but the design structure did not fully consider the three aspects of the day, week, and month, so it was not completely accurate in the prediction. In order to improve this situation, we add the day, week, and month convolution neural network (DWMCNN) to feature extraction in the framework of edge computing. DWMCNN is described in detail in Section 5.

5. Feature Extraction Based on DWMCNN

5.1. Convolution Neural Network (CNN). The CNN algorithm, as the feature extractor of the proposed model, trains the model parameters in the centralized data processing center and assigns them to the designated edge data center network, which is a kind of feed-forward neural network. This is an artificial neural network designed by simulating the structure of a cat's visual nerve as inspired by the structure of that nerve. The models developed from it, such as AlexNet, visual geometry group (VGG) network, and ResNet, are widely used in image processing [29].

The architecture of a CNN is composed of many distinct layers that transform input features into output features by differentiable functions. The basic convolution process of CNN is as follows. The convolution layer consists of a group of learnable filters or cores with small receptive fields but extends to the whole depth of the input volume. During the forward passage, each filter convolutes the input volume on the width and height of the filter, calculates the dot product

FIGURE 3: Comparison between normal users and thieves.

FIGURE 4: Monthly electricity consumption curve of normal users.

between the filter inlet and the input, and generates the two-dimensional activation map of the filter. The pooling layer is a form of nonlinear downsampling that is used to gradually reduce the space size of the representation and reduce the number of parameters and the amount of calculation in the network so as to control overfitting. After several convolutions and maximum pool layers, the high-level reasoning in the neural network is completed through the complete connection layer. The neurons in the fully connected layer are connected to all of the activation in the previous layer. The fully connected layer is used to generate the final output.

5.2. DWMCNN. To more comprehensively extract the features needed by users' electricity consumption data, the traditional CNN network structure needs to be improved.

FIGURE 5: Monthly electricity consumption curve of energy thieves.

The traditional LetNet5 network structure is simple and is composed of two convolution layers, two pooling layers, and two fully connected layers. The convolution kernel is 5×5 , stride=1, and the pooling layer uses max pooling. The potential power consumption relationship cannot be extracted effectively. In this paper, uncontrollable factors such as user lifestyle, seasonal change, and user type are considered when deciding the network structure, and the characteristics of user power consumption change are diverse. A CNN feature extraction framework for the day, week, and month is designed.

As shown in Figure 6, the DWMCNN framework is composed of 1D shape daily load feature convolution, 2D shape weekly load feature convolution, and monthly load feature convolution. We explain this in detail as follows:

- (1) *Daily Load Feature Extraction.* Daily load feature extraction is realized by a fully connected neural network layer. It learns global knowledge from 1D power consumption data. Customer electricity consumption is essentially 1D time series data. Each neuron in the full connectivity layer determines the output of the node according to the rectified linear unit (ReLU) activation function. The equation for ReLU performs as follows:

$$f(y) = \begin{cases} 0, & y < 0, \\ y, & y \geq 0, \end{cases} \quad (3)$$

where y is determined by the following equation:

$$y_j = \sum_{i=1}^n w_{i,j} x_i + b_1, \quad (4)$$

where y_j is the output of the complete connection layer of the j_{th} neuron, n is the length of one-dimensional input data, w_i is the neuron weight between the first input value and the j_{th} neuron, b is

FIGURE 6: CNN structure chart of daily, weekly, and monthly load feature extraction.

the neuron weight between the first input value, and the j_{th} neuron is the deviation. After calculation, the value to the connection unit is sent to the higher layer through the activation function to determine its contribution to the next prediction.

The input shape of one-dimensional daily load data is as follows:

$$M_{1 \times d} = [P_1 \ P_2 \ \dots \ P_d], \quad (5)$$

where d is the total number of days of historical power consumption data of users.

- (2) *Weekly and Monthly Load Feature Extraction.* Because the daily electricity consumption fluctuates in a relatively independent way, it is difficult to identify the periodicity or nonperiodicity of electricity consumption from one-dimensional electricity consumption data. If we analyze the power consumption data of several weeks together, we can easily identify the abnormal power consumption. Inspired by this observation, a deep CNN component is designed that transforms the two-dimensional data into two-dimensional data, convolutes the features, and combines them. For the input layer, we have two input threads: one thread is arranged weekly and the other thread is arranged monthly. After the double parallel convolution layer, the two groups of data are merged after they have the same shape. The input shapes of weekly and monthly load data are shown as follows:

$$M_{7 \times w} = \begin{bmatrix} P_{w1d1} & P_{w1d2} & \dots & P_{w1d7} \\ P_{w2d1} & P_{w2d2} & & P_{w1d7} \\ \vdots & & \ddots & \vdots \\ P_{wwd1} & P_{wwd1} & \dots & P_{wwd7} \end{bmatrix}, \quad (6)$$

$$M_{28 \times m} = \begin{bmatrix} P_{m1d1} & P_{m1d2} & \dots & P_{m1d28} \\ P_{m2d1} & P_{m2d2} & & P_{m2d28} \\ \vdots & & \ddots & \vdots \\ P_{mmd1} & P_{mmd2} & \dots & P_{mmd28} \end{bmatrix},$$

where w is the total number of power consumption weeks of the user's historical power consumption data. m is the total number of power consumption months of the user's historical power consumption data.

- (3) *Combination Extraction and Classification.* Combining feature extraction and one-dimensional convolution to extract daily load features and two-dimensional convolution to extract weekly and monthly load features, the weighted sum of their output is used as a hidden feature for combination and then through the full connection layer. Traditionally, a softmax classifier is used in the last output layer of CNN.

For the classification problem, the softmax function is a common function added to the output layer to obtain the category. The k -dimensional vector of any real value is compressed into the k -dimensional vector of the real value, where each entry is in the range of (0, 1), and all entries add up to 1.

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}, \quad f^0 y' = 1, \dots, K. \quad (7)$$

The classifier based on CNN-RF cancels the softmax classifier, outputs 32 dimensional features directly from the fully connected layer, and then predicts the categories. The RF classifier can be defined as follows:

$$\text{pred} = \text{sigm}(w_{\text{rf}} \cdot X_l + b_l), \quad (8)$$

where sigm is a sigmoid function that maps the outliers to 0 and the normal values to 1. The parameter set of the RF layer includes the number of decision trees and the maximum depth of the tree, which are obtained by a grid search algorithm.

To train the neural network, we define the loss function and optimizer to adjust the weight. In the neural network framework, we use classification cross entropy as the loss function and random gradient descent as the optimizer. The cross entropy of distributions u and v on a given discrete set is defined as follows:

$$H(u, v) = - \sum_x u(x) \log(v(x)). \quad (9)$$

Stochastic gradient descent (SGD) is an iterative method to optimize a differentiable objective function and a stochastic approximation of gradient descent optimization. The basic idea is to obtain a “gradient” through randomly selected data (x_i, y_i) to update the weight W via $W_{t+1} \rightarrow W_t + \eta \theta(-y_i W_i^T x_i)(y_i x_i)$

- (4) *Technique of Selecting Parameters and Avoiding Overfitting.* Table 1 summarizes the detailed parameters of the proposed DWMCNN structure, including the number of filters in each layer, filter size, and step size. Some units are randomly deleted from the neural network in the training process, which can prevent these units from adapting to each other too much and make a neuron independent of the existence of other specific neurons. The application of appropriate training methods can also help to reduce overtraining. Each iteration increases the weight, which is essentially a penalty. We also use binary cross entropy as the loss function. Finally, a grid search algorithm is used to optimize RF classifier parameters such as the maximum number of decision trees and features.

6. Implementation

To evaluate the performance of the proposed energy theft detection scheme based on the fact that other energy theft detection schemes are more realistic, an algorithm is implemented in Python 3.7, a CNN is implemented in the TensorFlow and Keras frameworks, and the interface

between the RF and CNN is implemented by the Scikit-learn module. The energy usage data comes from SGCC.

6.1. Data Preprocess. In SGCC data, due to various external factors, such as smart meter failure, unreliable measurement data, and unplanned system maintenance, error or null data inevitably appear in the dataset. Because the amount of data that cannot be read by the system for any reason will directly affect the effectiveness of the model, the preprocessing stage of the dataset is very important for the system.

6.1.1. Data Selection. In this study, data similar to the actual situation should be selected. Most of the electricity consumption data from 2014 and 2015 contain NaN and zero values, while the data in 2016 are more complete and contain fewer NaN data. Table 2 shows that in 2016, there were 169 users with 100–200 NaN data and 132 users with more than 200 NaN data. The number of users without any NaN and 0 data points was 30,341.

To keep as much data as possible, it is necessary to approximate the NaN value. First, according to the principle of 3 Sigma, the statistical error values due to meter failure and other reasons are removed. Then, according to equation (11), the daily power consumption of customers with NaN and zero data is eliminated.

$$f(x_{i,t}) = \begin{cases} \text{avg}(x_{i,t}) + 2\sigma(x_{i,t}), & |x_{i,t} - \text{avg}(x_{i,t})| > \theta, \\ x_{i,t}, & \text{else}, \end{cases} \quad (10)$$

where i is user i , $x_{i,t}$ is the daily power consumption data of the user on day t , $\text{avg}(x_{i,t})$ is the historical average daily power consumption of the user, $\sigma(x_{i,t})$ is the standard deviation of the user's historical daily electricity consumption, and θ is an artificially set deviation threshold.

$$f(x_{i,t}) = \begin{cases} \frac{x_{i,t-1} + x_{i,t+1}}{2}, & x_{i,t} \in \text{NaN}, x_{i,t+1} \text{ or } x_{i,t-1} \notin \text{NaN}, \\ 0, & x_{i,t} \in \text{NaN}, x_{i,t+1} \text{ or } x_{i,t-1} \in \text{NaN}, \\ x_i, & x_i \notin \text{NaN}, \end{cases} \quad (11)$$

where i is user i , $x_{i,t}$ is the daily power consumption data of the user on day t , and $x_{i,t-1}$, $x_{i,t+1}$ represents the daily power consumption data of the user on the day before and after day t respectively. If x_i is empty, then it is represented as NaN, which means that the missing value is uploaded by a smart meter.

To speed up the gradient descent to find the optimal solution and improve the accuracy, it is necessary to normalize the power consumption data. We choose the max-min scaling method to normalize the data according to the following equation:

TABLE 1: DWMCNN network structure.

Input layer		
Day input[1036 × 1 × 1]	Week input[148 × 7 × 1]	Month input[37 × 28 × 1]
dense_d[128]	conv_week_1[148 × 7 × 16]	conv_month_1[37 × 28 × 16]
	pool_week_1[146 × 5 × 16]	pool_month_1[35 × 26 × 16]
	conv_week_2[146 × 5 × 32]	conv_month_2[35 × 26 × 32]
	pool_week_2[36 × 4 × 32]	pool_month_2[33 × 24 × 32]
	conv_week_3[36 × 4 × 64]	conv_month_3[33 × 24 × 64]
	pool_week_3[9 × 3 × 64]	pool_month_3[11 × 8 × 64]
	conv_week_4[9 × 3 × 64]	conv_month_4[11 × 8 × 64]
	pool_week_3[3 × 2 × 64]	pool_month_4[3 × 2 × 64]
	concatenate_1[3 × 2 × 128]	
	flatten[768]	
	dense_wm[128]	
	concatenate_2[256]	
	dense_1[128]	
	dense_2[32]	
	dense_3[1]	

TABLE 2: Dataset summary.

Num of data	NaN	Zero	Normal	Rate (%)
<10	13.693	2.441	26.237	61.92
11–50	3.113	2.341	36.917	87.12
51–100	387	1.410	40.574	95.75
101–200	169	2.117	40.085	94.60
>200	132	6.023	36.216	85.47
Total	17.494	14.332	30.341	71.60

$$x_i = \frac{x_i - \min(x)}{\max(x) - \min(x)}. \quad (12)$$

6.2. Evaluation Method. Because it often expensive to check the identification of abnormal users, it is very important to predict abnormal users accurately. The confusion matrix is a basic tool to evaluate the performance of classifiers, as shown in Figure 7.

TP indicates that the predicted normal user is actually a normal user, and TN indicates that the predicted abnormal user is actually an abnormal user. The higher the TP and TN are, the higher the detection effect. FP is the predicted normal user, but the actual abnormal user, and FN means the predicted abnormal user but the actual normal user.

According to the confusion matrix, several evaluation indexes can be derived: accuracy (PR), recall (RE), F_1 score, etc.

$$\begin{aligned}
 \text{Pr} &= \frac{\text{TP}}{(\text{TP} + \text{FP})}, \\
 \text{Re} &= \frac{\text{TP}}{(\text{TP} + \text{FN})}, \\
 F_1 &= \frac{2 * (\text{Pr} * \text{Re})}{(\text{Pr} + \text{Re})}, \\
 \text{TPR} &= \frac{\text{TP}}{(\text{TP} + \text{FN})}, \\
 \text{TPR} &= \frac{\text{TP}}{(\text{TP} + \text{FN})}.
 \end{aligned} \quad (13)$$

TPR is the proportion of the number of normal users predicted by the detection model to all actual normal users, and FPR is the proportion of the number of abnormal users predicted by the detection model to all actual abnormal users.

The area under the receiver operating characteristic curve (AUC): By changing the threshold, a receiver operating characteristic (ROC) curve was drawn with TPR to FPR. The higher the AUC, the better the model can distinguish between abnormal and normal [30]. When AUC is 0.5, the model has no class separation ability.

6.3. Method Comparison. To evaluate the accuracy of the day, week, and month convolution neural network-random forest (DWMCNN-RF), nondeep learning methods including SVM, RF, gradient enhanced decision tree (GDBT), and logistic regression (LR) were used to carry out comparative experiments. In addition, we also compared the classification results of various supervised classifiers: CNN feature extraction and SVM classifier (CNN-SVM) and CNN feature extraction and GDBT classifier (CNN-GDBT). There were compared with the results of previous classification work. The following six methods were introduced and the results were analyzed:

LR: the basic model in binary classification, which is equivalent to a neural network with a sigmoid activation function. Any value greater than 0.5 is classified as normal mode, and any value less than 0.5 is classified as abnormal mode.

	Predicted normal user	Predicted abnormal user
Actual abnormal user	TP	FN
Actual normal user	FP	TN

FIGURE 7: Confusion matrix.

SVM: the classifier finds the optimal separation hyperplane by projecting the data into the feature space and transforming the nonlinear separable problem into a linearly separable problem.

GBDT: the model is an iterative decision tree algorithm composed of multiple decision trees, and the results of all trees are summed as the final result.

CNN: this uses a softmax classifier in the last layer of the network structure.

CNN-RF: this model uses the same feature extractor and classifier, but the network structure of the CNN is different from that of the DWMCNN, which uses the simplest LetNet5 structure.

DWMCNN-SVM/DWMCNN-GBDT: the neural network structure of these two models is the same as that of DWMCNN, but different classifiers are used.

The classification results of the proposed detection model on the test set are as follows: 935 samples for TP, 134 samples for FN, 22 samples for FP, and 578 samples for TN. Therefore, according to the equation, the accuracy, precision, and F_1 score of the proposed model are all 0.97, as shown in Table 3.

Class 0 is the exception user class, and Class 1 is the normal user class. The ROC curve of the DWMCNN-RF model was drawn as shown in Figure 8. The AUC value was 0.988, which was much better than that of the baseline model ($AUC = 0.5$). This shows that the algorithm can classify these two classes accurately.

The parameters of the comparison method are summarized in Table 4 and experiments were carried out accordingly. The results of different methods are shown in Figure 8, which shows the AUC of DWMCNN-RF, DWMCNN-GDBT, DWMCNN-SVM, CNN, SVM, CNN-RF, LR, and GDBT.

The results of different methods are shown in Figure 9. The AUC values of DWMCNN-RF, DWMCNN-GBDT, DWMCNN-SVM, CNN, SVM, CNN-RF, LR, and GDBT are 0.99, 0.98, 0.98, 0.92, 0.76, 0.93, 0.62, and 0.77 respectively.

Figure 10 shows the results of all comparative experiments in terms of accuracy, recall, and F_1 score. Among the eight different detection algorithms, deep learning

TABLE 3: Classification score summary.

	Precision	Recall	F_1 score
Class 0	0.96	0.96	0.96
Class 1	0.97	0.98	0.97
Average	0.97	0.97	0.97

— AUC (DWMCNN-RF) = 0.99

FIGURE 8: ROC curve for the paper model.

TABLE 4: Parameter summary.

Methods	Input data	Parameters
LR	Features (1D)	Penalty: L2 Inverse of regulation strength: 1.0 Penalty parameter of the error term: 105
SVM	Features (1D)	Parameter of kernel function (RBF): 0.0005
GBDT	Features (1D)	The number of estimators: 200
RF	Features (1D)	The number of trees in the forest: 100
CNN	Features (2D)	The max depth of each tree: 30

(including improved CNN network structure and ordinary CNN network structure) outperforms machine learning (such as LR, GBDT, and SVM). For deep learning methods, the algorithm using a CNN network structure in this paper outperforms the algorithm not using a CNN network structure in this paper (comparison of DWMCNN-RF with CNN-RF and CNN). Among the algorithms using the network structure in this paper, RF is the best classifier (comparison of DWMCNN-RF with DWMCNN-SVM and DWMCNN-GDBT).

The reason for the above results is that compared with the classical machine learning method, deep learning does not need feature engineering. Classical machine learning algorithms usually require complex feature engineering. First, deep exploratory data analysis is performed on the dataset, and then a simple dimensionality reduction process is performed. Finally, the best function must be carefully selected to pass on to the machine learning algorithm. When

FIGURE 9: ROC curves for comparative algorithms.

FIGURE 10: Performance of comparative algorithms.

using deep networks, we do not need to do this because we can usually achieve good performance by simply passing data directly to the network. In a deep learning network, the DWCNN is better than a CNN because the DWCNN has periodic characteristics for daily, weekly, and monthly

data of power data and can extract features more effectively. In addition, to further demonstrate the classification performance of the proposed method, a confusion matrix heat map of the proposed method and the ordinary CNN structural feature extraction method are shown in Figure 11.

FIGURE 11: Heat map of confusion matrix of general CNN network structure (a) and proposed DWM network structure (b).

The heat map of the confusion matrix shows that the CNN method easily decomposes normal data into abnormal data without the deep learning method of an improved structure, and it is not robust to normal load changes. In the selection of classifiers, the RF classifier and the model proposed in this paper have the best combination effect, which is suitable for large-scale training samples and high-dimensional feature data.

7. Conclusion

Based on a power consumption information acquisition system, this paper proposed an energy theft detection method for an edge data center. This method includes clustering and CNN training at a centralized data center, feature extraction based on a CNN feature extractor at an edge data center, and RF algorithm training based on the extracted features. The advantage of this method was proven in the following experiments:

- (1) Using K -means clustering technology can greatly shorten the computing time and realize distributed data processing. Compared with the traditional method of processing power consumption data at a centralized data center, this method has the advantages of fast calculation speed, less bandwidth occupation, and good privacy protection.
- (2) Compared with the principal components analysis (PCA) based feature extraction method, the improved CNN network feature extraction method proposed in this paper can effectively find the periodicity of the data, which is consistent with the daily, weekly, and monthly variation characteristics of power consumption data. Compared with other traditional classifiers, the DWMCNN-RF combination model

has higher accuracy and better robustness and can effectively realize energy thief detection.

In future research, we will continue to improve the monitoring function under this framework. User load forecasting and anomaly recognition based on edge computing is an important direction of framework development.

Data Availability

This dataset released by State Grid Corporation of China (SGCC) contains the electricity consumption data of 42,372 electricity customers within 1,035 days (<https://www.sgcc.com.cn/>).

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Science and Technology Project of Metering Centre of Sichuan Electric Power Corporation: Research on New Electric Energy Information Interaction Equipment Based on Holographic Sensing and Edge Computing (52199719001M).

References

- [1] P. Jokar, N. Arianpoo, and V. C. M. Leung, "Electricity theft detection in AMI using customers' consumption patterns," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 216–226, 2015.
- [2] G. Bedi, G. Kumar Venayagamoorthy, R. Singh, R. Brooks, and K. C. Wang, "Review of Internet of Things (IoT) in

- electric power and energy systems," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 847–870, 2018.
- [3] Y. Saleem, N. Crespi, M. H. Rehmani, and R. Copeland, "Internet of Things-aided smart grid: technologies, architectures, applications, prototypes, and future research directions," *IEEE Access*, vol. 7, pp. 62962–63003, 2019.
 - [4] T. Ahmad, D. Q. U. Hasan, and S. Zada, "Non-technical loss detection prevention and suppression issues for AMI in smart grid," *International Journal of Scientific and Engineering Research*, vol. 6, no. 3, pp. 217–228, 2015.
 - [5] R. Amudhevali and T. Sivakumar, "IoT based smart energy metering system for monitoring the domestic load using PLC and SCADA," in *Proceedings of the International Virtual Conference on Robotics, Automation, Intelligent Systems and Energy (IVC RAISE 2020)*, IOP Publishing, Erode, India, December 2020.
 - [6] A. A. Cárdenas, S. Amin, G. Schwartz, R. Dong, and S. Sastry, "A game theory model for electricity theft detection and privacy-aware control in AMI systems," in *Proceedings of the 2012 50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1830–1837, Monticello, IL, USA, October 2012.
 - [7] C. H. Park and T. Kim, "Energy theft detection in advanced metering infrastructure based on anomaly pattern detection," *Energies*, vol. 13, no. 15, p. 3832, 2020.
 - [8] J. Li, Y. Yang, and J. Stella Sun, "SearchFromFree: adversarial measurements for machine learning-based energy theft detection," in *Proceedings of the 2020 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pp. 1–6, Tempe, AZ, USA, November 2020.
 - [9] S. Shekara Sreenadh Reddy Depuru, L. Wang, and V. Devabhaktuni, "Support vector machine based data classification for detection of electricity theft," in *Proceedings of the 2011 IEEE/PES Power Systems Conference and Exposition*, pp. 1–8, Phoenix, AZ, USA, March 2011.
 - [10] J. Nagi, K. S. Yap, S. K. Tiong, S. K. Ahmed, and M. Mohamad, "Nontechnical loss detection for metered customers in power utility using support vector machines," *IEEE Transactions on Power Delivery*, vol. 25, no. 2, pp. 1162–1171, 2009.
 - [11] J. Nagi, K. S. Yap, S. K. Tiong, S. K. Ahmed, and F. Nagi, "Improving SVM-based nontechnical loss detection in power utility using the fuzzy inference system," *IEEE Transactions on Power Delivery*, vol. 26, no. 2, pp. 1284–1285, 2011.
 - [12] Y. Liu, R. Yuan, S. Zheng, K. Yan, and H. Miao, "An abnormal detection of positive active total power based on local outlier factor," in *Proceedings of the 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA)*, pp. 180–183, Shenyang, China, January 2021.
 - [13] R. Razavi, A. Gharipour, M. Fleury, and I. J. Akpan, "A practical feature-engineering framework for electricity theft detection in smart grids," *Applied Energy*, vol. 238, pp. 481–494, 2019.
 - [14] P. Glauner, A. Boechat, L. Dolberg et al., "Large-scale detection of non-technical losses in imbalanced data sets," in *Proceedings of the 2016 IEEE Power and Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, Minneapolis, MN, USA, 2016.
 - [15] M. Adil, N. Javaid, U. Qasim, I. Ullah, M. Shafiq, and J.-G. Choi, "LSTM and bat-based RUSBoost approach for electricity theft detection," *Applied Sciences*, vol. 10, no. 12, p. 4378, 2020.
 - [16] P. Glauner, J. Augusto Meira, L. Dolberg, R. State, F. Bettinger, and Y. Rangoni, "Neighborhood features help detecting non-technical losses in big data sets," in *Proceedings of the 3rd IEEE/ACM International Conference on Big Data Computing, Applications and Technologies*, pp. 253–261, Chengdu, China, 2016.
 - [17] V. Hodge and J. Austin, "A survey of outlier detection methodologies," *Artificial Intelligence Review*, vol. 22, no. 2, pp. 85–126, 2004.
 - [18] B. C. Costa, B. L. A. Alberto, A. M. Portela, W. Maduro, and O. Eler, "Fraud detection in electric power distribution networks using an ANN-based knowledge-discovery process," *International Journal of Artificial Intelligence and Applications*, vol. 4, no. 6, pp. 17–23, 2013.
 - [19] J. I. Guerrero, C. León, I. Monedero, F. Biscarri, and J. Biscarri, "Improving knowledge-based systems with statistical techniques, text mining, and neural networks for non-technical loss detection," *Knowledge-Based Systems*, vol. 71, pp. 376–388, 2014.
 - [20] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: a strong baseline," in *Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 1578–1585, Anchorage, AK, USA, 2017.
 - [21] S. S. S. R. Depuru, L. Wang, V. Devabhaktuni, and R. C. Green, "High performance computing for detection of electricity theft," *International Journal of Electrical Power and Energy Systems*, vol. 47, pp. 21–30, 2013.
 - [22] M. N. Hasan, R. N. Toma, A.-A. Nahid, M. M. M. Islam, and J.-M. Kim, "Electricity theft detection in smart grid systems: a CNN-LSTM based approach," *Energies*, vol. 12, no. 17, p. 3310, 2019.
 - [23] M. A. de Souza, J. L. R. Pereira, G. D. O. Alves, B. C. de Oliveira, I. D. Melo, and P. A. N. Garcia, "Detection and identification of energy theft in advanced metering infrastructures," *Electric Power Systems Research*, vol. 182, Article ID 106258, 2020.
 - [24] S.-C. Yip, K. Wong, W.-P. Hew, M.-T. Gan, R. C.-W. Phan, and S.-W. Tan, "Detection of energy theft and defective smart meters in smart grids using linear regression," *International Journal of Electrical Power and Energy Systems*, vol. 91, pp. 230–240, 2017.
 - [25] C. Tian, J. Ma, C. Zhang, and P. Zhan, "A deep neural network model for short-term load forecast based on long short-term memory network and convolutional neural network," *Energies*, vol. 11, no. 12, p. 3493, 2018.
 - [26] A. Capozzoli, F. Lauro, and I. Khan, "Fault detection analysis using data mining techniques for a cluster of smart office buildings," *Expert Systems with Applications*, vol. 42, no. 9, pp. 4324–4338, 2015.
 - [27] E. W. S. Angelos, O. R. Saavedra, O. A. C. Cortés, and A. N. de Souza, "Detection and identification of abnormalities in customer consumption in power distribution systems," *IEEE Transactions on Power Delivery*, vol. 26, no. 4, pp. 2436–2442, 2011.
 - [28] Z. Zheng, Y. Yang, X. Niu, H.-N. Dai, and Y. Zhou, "Wide and deep convolutional neural networks for electricity-theft detection to secure smart grids," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1606–1615, 2017.
 - [29] A. Ullah, N. Javaid, O. Samuel, M. Imran, and M. Shoaib, "CNN and GRU based deep neural network for electricity theft detection to secure smart grid," in *Proceedings of the 2020 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 1598–1602, Imassol, Cyprus, 2020.
 - [30] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proceedings of the 23rd International Conference on Machine Learning*, pp. 233–240, Pittsburgh, PA, USA, June 2006.